|  |
| --- |
| **电化学开发板** |
| **模拟量输出**  **设计指南** |
| **Ver1.0** |



目录

[一、产品简介 3](#_Toc15638)

[二、具体参数 3](#_Toc29526)

[三、工作原理图 3](#_Toc2765)

[四、编程和标定思路 6](#_Toc16080)

[五、开发板外观图 7](#_Toc25408)

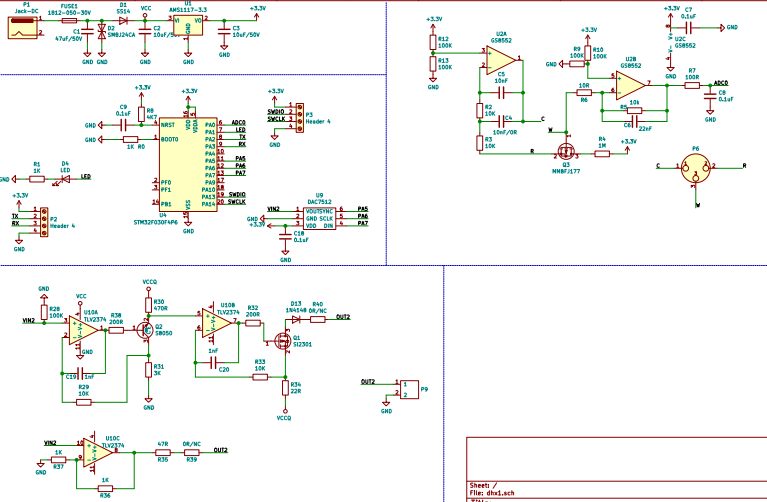
**一、产品简介**

此开发板是我司针对于2电极和3电极电化学探头检测不同的气体含量而开发。此开发板支持4-20mA输出、0-5V输出、0-10V输出，供电方式支持5/12V。此开发板是基于STM32F030F4P6芯片，采用精密运放芯片GS8552采集电化学探头的输出信号，经过算法处理，可以用不同的方式输出。我司提供开发例程，模块，客户可以通过更改部分电路以及例程实现相应的功能。

**二、具体参数**

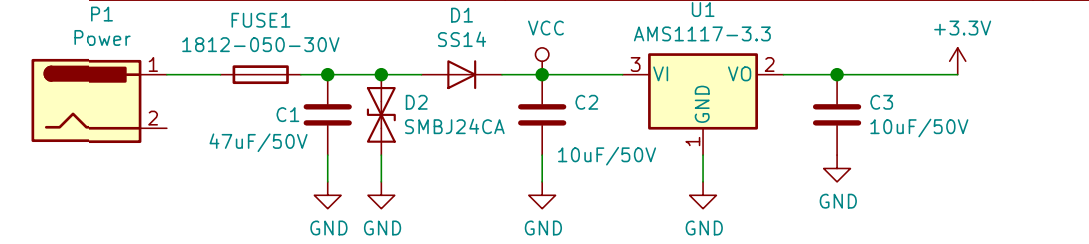
|  |  |
| --- | --- |
| 参数 | 技术指标 |
| 输入电源 | 5V/12V DC |
| 输出方式 | 4-20mA/0-5V/0-10V输出 |
| 尺寸 | 82mm\*35mm\*16mm(长\*宽\*高) |
| 储存温度 | -25OC - +85OC |
| 工作温度 | -10OC - +70OC |
| 重量 | 0.05kg |

1. **工作原理图**



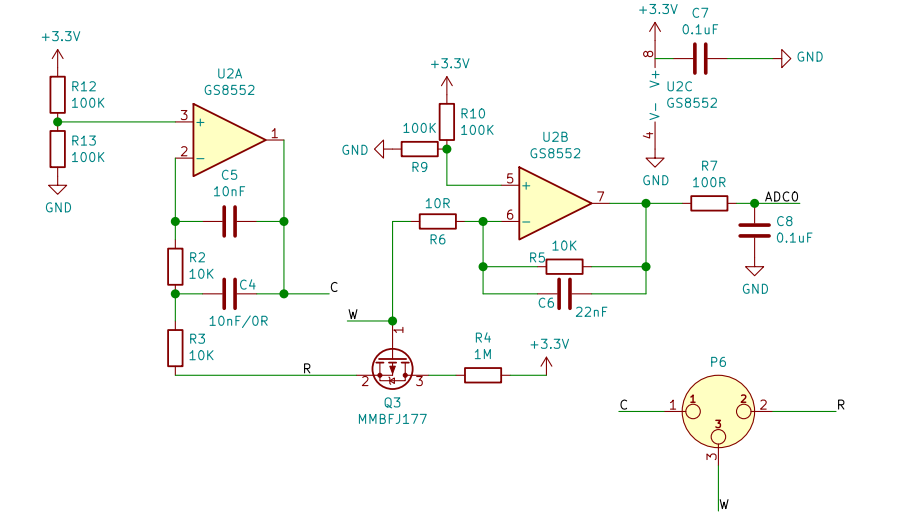
图一 总原理图

如图一所示，这是总原理图，包含降压模块、ADC采集模块、STM32F0303F4P6单片机最小系统模块、485模块、232模块。



图二 电源模块

如图二所示，这是基于AMS1117-3.3V的降压电路，可以把输入为5V或者输入为12V 的电压转换成3.3V输出。



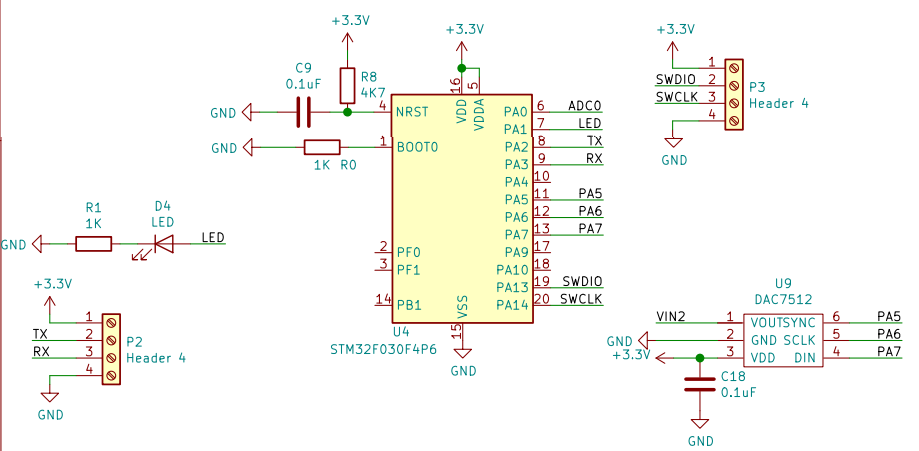
图三 ADC采集模块

如图三所示是ADC采集模块，使用的是GS8552运放芯片，该GS8552放大器是单电源，微功率，零漂移CMOS运算放大器，放大器提供带宽为1.5MHz，轨对轨输入和输出，单电源操作从2.1V到5.5V。本采集模块支持2电极和3电极电化学气体探头。C是对电极，W是工作电极，R是参比电极。

关于2电极和3电极的修改说明

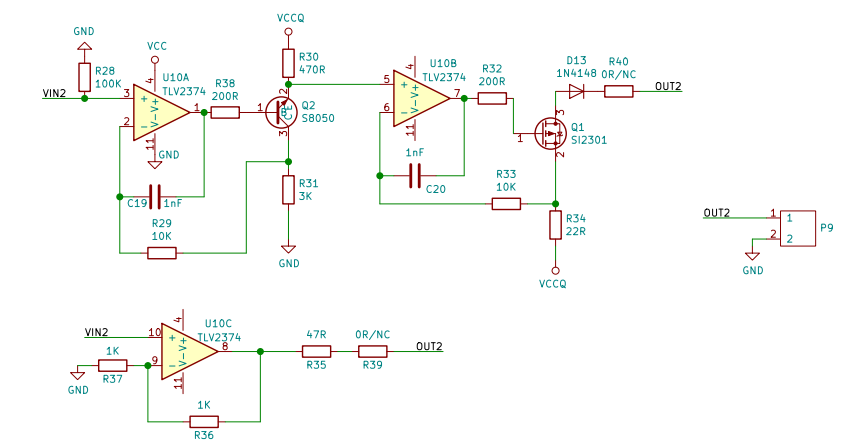
1）若使用2电极电化学探头，C4应焊接0R电阻，R3应焊接100R电阻；

2）若使用3电极电化学探头，C4应焊接10nF电容，R3应焊接10K电阻。



图四 单片机模块

如图四所示，这款单片机是STM32F030F4P6，它的工作电压是2.4V-3.6V，封装是TSSOP，有20个引脚，最大频率是48MHz，具有各种增强型外设和 I/O。 有如 I2C、SPI 和 USART 等通信接口，以及 12 位 ADC、16 位计时器和一个高级控制 PWM 计时器。用到的芯片还有DAC7512，单片机通过模拟SPI通信接口与7512芯片通信，控制其输出数字电压值，经差动缩放电路、电流/电压变换电路和功率驱动电路，最后输出恒定电流或者恒定电压。



图五 模拟量输出模块

如图五所示，是基于TLV2374的运放输出电路，从DAC7512芯片输出的电压值经过电路变化输出电流或者电压。

使用4-20ma电流输出和0-5V/0-10V电压输出，模拟量模块改板说明

1）使用4-20ma电流输出时，R40需焊接0R电阻，R39不焊，供电电源5V/12V均可；

2）使用0-5V电压输出时，R40不需要焊接，R39焊接0R电阻，此时供电电源5V/12V均可，可通过调换R36的阻值改变输出电压；

3）使用0-10V电压输出时，R40不需要焊接，R39焊接0R电阻，此时供电电源应为12V，可通过调换R36的阻值改变输出电压。

**四、编程和标定思路**

1. 编程思路

1）adc采集

一路adc采集，单片机处理采集到的数据，对采集到的数据进行软件滤波处理，得到一个较为稳定的数值。

1. 输出方式

输出方式为模拟量输出，如4-20ma，0-5V，0-10V，模拟量的值可以通过修改DAC7512(u16 DAData)的DAData值来改变。

1. 标定思路

模拟量和检测气体的浓度有一定的线性关系，根据线性关系，进行零点和满点标定。

1. 参考程序

1）adc采集处理程序

#define A 101

u32 caiji[A];

uint32\_t ADC\_Check(void)//采集处理程序

{

uint16\_t i,j,k,h;

u32 ResultI2;

for(i=0;i<A;i++)

{

while(!ADC\_GetFlagStatus(ADC1, ADC\_FLAG\_EOC ));//等待转换结束

caiji[i]=(uint32\_t)ADC\_ConvertedValue[i];

for(j=0;j<500;j++)

{;}

}

for(j=0;j<A-1;j++) //排序取中值

{

for(k=0;k<A-j-1;k++)

{

if(caiji[k]>caiji[k+1])

{

ResultI2=caiji[k];

caiji[k]=caiji[k+1];

caiji[k+1]=ResultI2;

}

}

}

ResultI=caiji[(A-1)/2];

ResultI=ResultI\*3300;

ResultI=ResultI/4096;

ADC\_ClearITPendingBit(ADC1,ADC\_IT\_EOC);

ResultI1 = ResultI;

return ResultI1;

}

2）DAC7512驱动程序

#define IO\_SYNC\_H GPIO\_SetBits(GPIOA,GPIO\_Pin\_5)

#define IO\_SCLK\_H GPIO\_SetBits(GPIOA,GPIO\_Pin\_6)

#define IO\_DIN\_H GPIO\_SetBits(GPIOA,GPIO\_Pin\_7)

#define IO\_SYNC\_L GPIO\_ResetBits(GPIOA,GPIO\_Pin\_5)

#define IO\_SCLK\_L GPIO\_ResetBits(GPIOA,GPIO\_Pin\_6)

#define IO\_DIN\_L GPIO\_ResetBits(GPIOA,GPIO\_Pin\_7)

void SetSDAPP()

{

GPIO\_InitTypeDef GPIO\_InitStructure;

GPIO\_InitStructure.GPIO\_Pin=GPIO\_Pin\_5|GPIO\_Pin\_6|GPIO\_Pin\_7;

GPIO\_InitStructure.GPIO\_Mode = GPIO\_Mode\_OUT;

GPIO\_InitStructure.GPIO\_OType = GPIO\_OType\_PP;

GPIO\_InitStructure.GPIO\_Speed=GPIO\_Speed\_50MHz; //高速GPIO

GPIO\_Init(GPIOA,&GPIO\_InitStructure);

GPIO\_SetBits(GPIOA,GPIO\_Pin\_7);

}

void SetSDAOD()

{

GPIO\_InitTypeDef GPIO\_InitStructure;

GPIO\_InitStructure.GPIO\_Pin=GPIO\_Pin\_7;

GPIO\_InitStructure.GPIO\_Mode = GPIO\_Mode\_OUT;

GPIO\_InitStructure.GPIO\_OType = GPIO\_OType\_OD;

GPIO\_InitStructure.GPIO\_Speed=GPIO\_Speed\_50MHz; //高速GPIO

GPIO\_Init(GPIOA,&GPIO\_InitStructure);

}

void DAC7512(u16 DAData) //满电压是5000mV 4096分度值

{

u8 i;

u32 temp=DAData;

SetSDAPP();

IO\_SCLK\_L;

IO\_SYNC\_L;//启动转换

IO\_SCLK\_H;

temp=temp\*5000/4096;// /5000\*4096输出电压值0-3.3V

if(temp>4095)

temp=4095 ;

DAData=temp;

for(i=0;i<16;i++)//AD7888写入命令与获取数值一次。

{

if((DAData&(0x8000>>i))>>(15-i))

{

IO\_DIN\_H;//启动转换

}

else

{

IO\_DIN\_L;//启动转换

}

IO\_SCLK\_L; //下降写入

IO\_DIN\_L;//启动转换;//每次数据移入后，DIN还需要置0真变态

IO\_SCLK\_H;//

}

IO\_SCLK\_L; // 这里还需要SCLK置0，更变态,datasheet上却没提

IO\_SYNC\_H;//下次转换必须先置1

SetSDAOD();

}

3）标定程序

void Tiaozheng()

{

dianliu=ResultI1;//\*\*ADCcheng/ADCchu+ADCjia;

if(dianliu\_0<=dianliu<=dianliu\_1)

{

dianliutrue=I1\*(dianliu-dianliu\_0)/(dianliu\_1-dianliu\_0);

}

else if(dianliu\_1<dianliu)

{

dianliutrue=I1\*(dianliu-dianliu\_1)/(dianliu\_1-dianliu\_0)+I1;

}

}

4）主程序

void main()//主程序

{

ReadInf();

Systick\_Init();

InitALL();

GPIO\_SetBits(GPIOA,GPIO\_Pin\_1);//点亮状态灯

while (1)

{

ADC\_Check(); //模拟量采集

Tiaozheng();

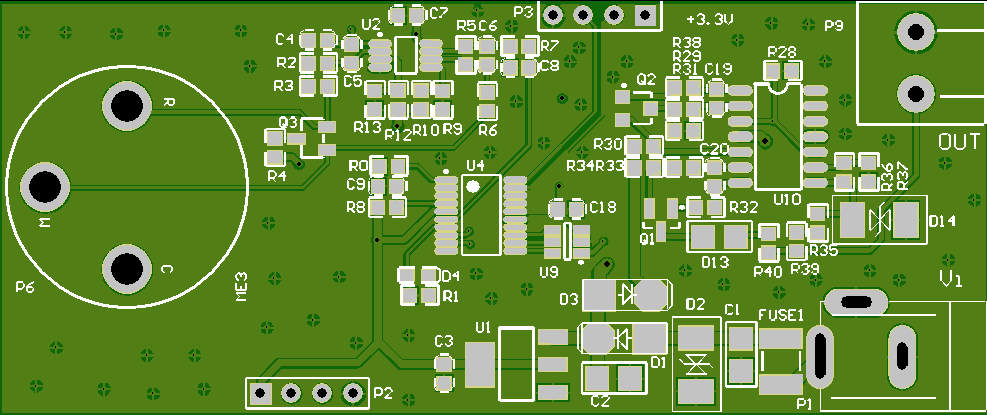
DAC7512(dianliutrue)

delay\_ms(100);

}

}

**五、开发板外观图**

图六 开发板PCB图